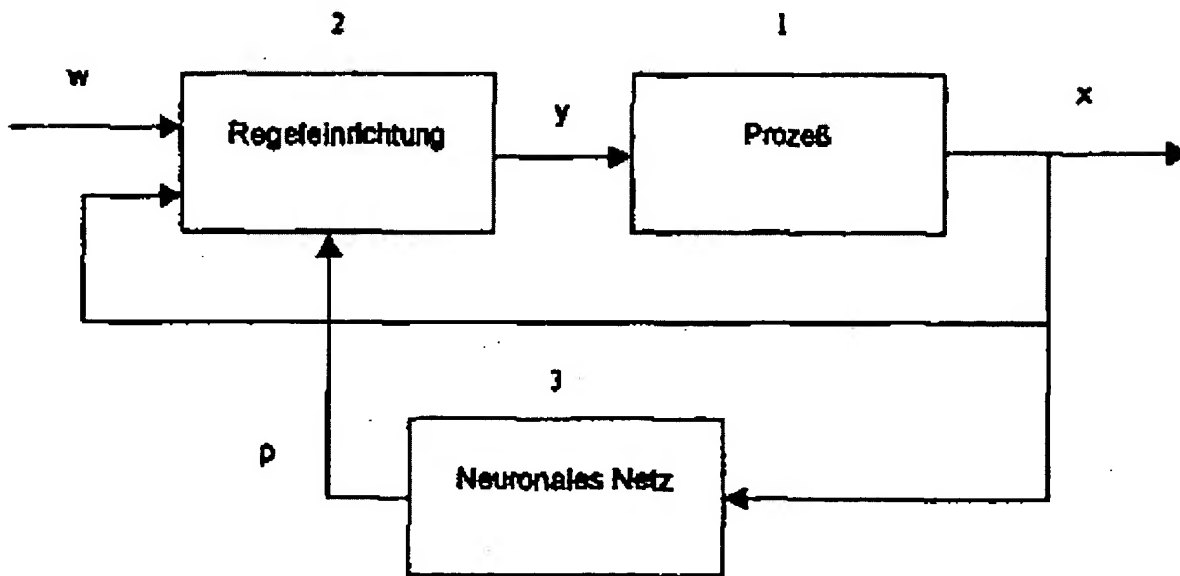


AN: PAT 1999-289088
TI: Self-adjusting device for process control system
PN: **DE19748718-A1**
PD: 06.05.1999
AB: NOVELTY - The device includes a neural network (3) which derives parameters for the control system based on a control parameter (x). As a basis for the neural self-adjustment, a test function is used as an adjustment parameter when the control system is switched off, or as a guide parameter when the control system is switched on. The detected response of the control parameter to the test function is fed directly or in converted form to the neural network.; USE - For controlling industrial automation processes. ADVANTAGE - The device improves the start-up and maintenance of control loops.
DESCRIPTION OF DRAWING(S) - neural network 3 control parameter x
x
PA: (PHIL/) PHILIPPSEN H W;
IN: PHILIPPSEN H W;
FA: **DE19748718-A1** 06.05.1999;
CO: DE;
IC: G05B-011/42; G05B-013/02;
MC: T01-J16C1; T06-A05A; T06-A06A9;
DC: T01; T06;
FN: 1999289088.gif
PR: DE1048718 05.11.1997;
FP: 06.05.1999
UP: 14.06.1999



BEST AVAILABLE COPY

THIS PAGE BLANK (USPTO)

02 P 04668



B3(3)

19 BUNDESREPUBLIK
DEUTSCHLAND



DEUTSCHES
PATENT- UND
MARKENAMT

Offenlegungsschrift
10 DE 197 48 718 A 1

51 Int. Cl.⁶:
G 05 B 13/02
G 05 B 11/42

21 Aktenzeichen: 197 48 718.1
22 Anmeldetag: 5. 11. 97
43 Offenlegungstag: 6. 5. 99

DE 197 48 718 A 1

71 Anmelder:
Philippsen, Hans Werner, Dr.-Ing., 28215 Bremen,
DE

72 Erfinder:
gleich Anmelder

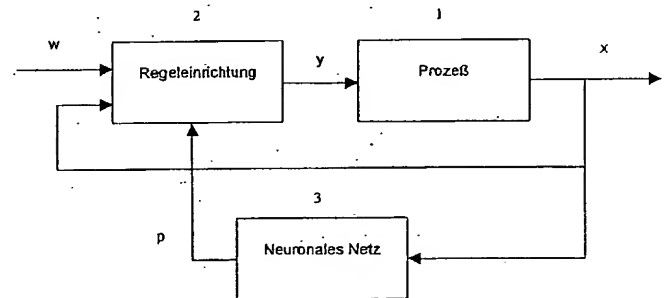
Die folgenden Angaben sind den vom Anmelder eingereichten Unterlagen entnommen

54 Regler - Selbsteinstellungseinrichtung auf der Grundlage Neuronaler Netze

57 Die Erfindung betrifft die Selbsteinstellung einer Automatisierungseinrichtung für einen Prozeß (1), wobei mindestens eine Führungsgröße (w) und mindestens eine Ausgangsgröße des Prozesses (1) als Regelgröße (x) auf die Regeleinrichtung geführt sind und die Selbsteinstellung der Regeleinrichtung (Parametervektor p) durch ein Neuronales Netz (3) in Abhängigkeit von der Regelgröße (x) erfolgt.

Das Neuronale Netz wird anhand eines Lerndatensatzes trainiert. Eine Testfunktion, die als Stellgröße (y) auf den Prozeß (1) bei ausgeschalteter Regeleinrichtung oder als Führungsgröße (w) bei eingeschalteter Regeleinrichtung, verursacht eine dynamische Änderung der Regelgröße (x), die die Grundlage für die neuronale Selbsteinstellung ist. Der Vorteil besteht in einer schnellen und robusten Reglereinstellung hoher Güte, die mit geringen Anforderungen an Rechenleistung und Speicherkapazität verbunden ist.

Die Erfindung findet Verwendung in der Prozeßautomatisierung.



DE 197 48 718 A 1

Die nachfolgende Beschreibung erfolgt für eine konkrete industrietypische Regeleinrichtung, dem PID-Regler und für eine sprungförmige Testfunktion, die als Stellgröße bei ausgeschalteter Regeleinrichtung verwendet wird. Darüber hinaus findet ein Reglerentwurf nach dem Betragsoptimum statt.

Voraussetzung für einen optimalen Reglerentwurf ist die Streckenkenntnis in Form eines ausreichend genauen mathematischen Modells. Die Erarbeitung eines mathematischen Streckenmodells und der systematische Reglerentwurf sind zeitaufwendige Tätigkeiten mit großen Anforderungen an das Personal und daher finanziell nur selten vertretbar. Die Inbetriebnahme von Regelkreisen erfolgt daher meistens nicht auf der Grundlage eines mathematischen Modells und eines systematischen Reglerentwurfes. Einstellregeln sind sicherlich häufiger im Gebrauch, jedoch haben die Reglereinstellung auf Basis von Erfahrungen und Probiervorfahren die größte Einsatzhäufigkeit. Die Folge sind unter Umständen schlecht eingestellte Regler oder eine zeitaufwendige Inbetriebnahme. In beiden Fällen entstehen unnötige Kosten. Es besteht aus diesen Gründen in der Industrie ein aktueller Bedarf an einfachen und leistungsfähigen Einstellverfahren, die ein rasches, kostengünstiges Inbetriebnahme von PID-Reglern ermöglichen [4][5][3].

Neben den Einstellverfahren sind in der Vergangenheit neue Regelungskonzepte entwickelt worden, die eine Selbsteinstellung beinhalten.

Adaptive Regler, die selbsttätig ein mathematisches Abbild der Strecke erzeugen, haben sich jedoch trotz großer Anstrengungen nicht in der industriellen Praxis etablieren können. Gründe für die geringe Verbreitung adaptiver Regler sind u. a. die Inbetriebnahme, Stabilitätsprobleme im Regelkreis und die Kompliziertheit der Bedienung [12].

Auch neuere Ansätze, wie z. B. Fuzzy oder Neuronale Regler, entbinden den Regelungstechniker nicht von der Aufgabe, ein qualitatives Prozeßmodell zu erstellen bzw. genügend Datenmaterial zusammenzustellen. Die Erarbeitung von Regeln und Zugehörigkeitsfunktionen für den Fuzzy Regler ist im Vergleich mit dem konventionellen Reglerentwurf nicht schneller durchführbar und führt grundsätzlich nicht zu besseren Betriebsergebnissen [13]. Der zeitliche Aufwand, geeignete Trainingsdaten für einen Neuronalen Regler zusammenzustellen, die garantiert jeden Betriebsfall wiedergeben, sollte nicht unterschätzt werden. Die Vielzahl Neuronaler Netze und Regler sowie die dazugehörigen Trainingsverfahren stellen größere Anforderungen an Inbetriebnehmer und Bedienpersonal.

Der Funktionsumfang eines modernen digitalen PID-Reglers beinhaltet häufig eine Inbetriebnahmehilfe in Form einer Adapt-Taste. Die Anwender von PID-Reglern profitieren also von den Arbeiten zur adaptiven Regelung, wobei der PID-Regler weiterhin einfach bedienbar bleibt und seine bewährten Eigenschaften behält.

Das Inbetriebnahmewerkzeug SIEPID [12] verwendet ein praxisnahes Selbsteinstellungsverfahren. Dort werden mit einem robusten Identifikationsverfahren, ein Ersatzstreckenmodell anhand der Übergangsfunktion der Strecke bestimmt und mittels Betragsoptimum die Reglerparameter berechnet. Dieses Verfahren ist auch für Kompaktregler geeignet [11].

Einen prinzipiell ähnlichen Ansatz, einen PI-Regler mit Hilfe einer modernen Methode zu ergänzen, wird von Pfeiffer [8] vorgeschlagen. Die Reglerparameter werden – ausgehend von der Übergangsfunktion des geschlossenen – Regelkreises von einem Fuzzy System schrittweise ermittelt. Eingangsgrößen des Fuzzy Systems sind lediglich die Überschwingweite und das Verhältnis von An- und Ausregelzeit. Ausgangsgrößen sind die vorgeschlagenen Änderungen der Reglerparameter. Nach ca. 5 Sprungversuchen ist der PI-Regler eingestellt.

In dieser Arbeit soll ein Verfahren vorgestellt werden, das die Parameter eines PID-Reglers mit Hilfe eines Neuronalen Netzes entwirft. Grundlage sind Zeitprozentkennwerte der Strecke. Bild 1 zeigt die Struktur des Selbsteinstellungsverfahrens. Das Neuronale Netz erhält 17 Zeitprozentkennwerte als Eingangsgrößen. Daraus erzeugt das Neuronale Netz eine Abbildung auf die Reglerparameter, wobei die Parameter durch das Betragsoptimum [2] vorgegeben sind. Die Identifikation der Strecke und der Reglerentwurf erfolgen in einem Schritt. Es handelt sich um einen prozeßmodellfreien Reglerentwurf nach dem Betragsoptimum.

Anwendungszweck dieses Verfahrens ist die Unterstützung der Inbetriebnahme und Wartung von Regelkreisen. Das hier vorgestellte Inbetriebnahmewerkzeug eignet sich aufgrund der kleinen Netzgröße (5 Neuronen in der Zwischenschicht) für eine Integration in ein Automatisierungssystem (SPS) oder einen Kompaktregler. Der Einsatz von Neuronalen Netzen bietet den Vorteil, aufgrund der generalisierenden Eigenschaften, die Reglerinbetriebnahme auch auf der Grundlage gestörter Übergangsfunktionen durchführen zu können. Die Dauer der Reglereinstellung ist sehr kurz und unabhängig von Störungen. Das neuronale Inbetriebnahmewerkzeug benötigt geringe Ressourcen innerhalb eines Automatisierungsgerätes. Daher kann auf einen zusätzlichen Inbetriebnahme-PC verzichtet werden.

Neuronale Netze

Die im Rahmen dieser Arbeit verwendeten Neuronalen Netze gehören zu der Klasse der Multi Layer Perceptrons [1]. Die Neuronen, als kleinste Einheit, sind in Schichten zusammengefaßt, wobei die Neuronen innerhalb einer Schicht keinerlei Verbindung haben. Der Informationsfluß im Netz erfolgt von der Eingangsschicht zur Ausgangsschicht (Feed Forward Netz). Bild 2 zeigt ein Netz mit einer Zwischenschicht (hidden layer). Dieser Netztyp findet in dieser Arbeit Verwendung, wobei zusätzlich ein Bias zugeführt wird (siehe Bild 3).

Die Neuronen einer Schicht haben grundsätzlich den gleichen Aufbau, wie prinzipiell Bild 3 verdeutlicht.

Hervorzuheben sind neben den gewichteten Eingängen und dem Bias die Aktivierungsfunktionen am Ausgang. Im Falle eines zweischichtigen Netzes mit nichtlinearen sigmoidalen Aktivierungsfunktionen ist ein Neuronales Netz in der Lage, jede physikalische Funktion beliebig genau anzunähern (universal approximator) [1].

Sigmoidfunktionen sind nichtlineare differenzierbare Funktionen wie z. B. die Fermi Funktion mit einem Wertebereich von 0...1

$$f(s) = \frac{1}{1 + e^{-s}} \quad (1)$$

oder der Tangens hyperbolicus

$$f(s) = \frac{e^s - e^{-s}}{e^s + e^{-s}} = \frac{2}{1 + e^{-2s}} - 1 \quad (2)$$

mit einem Wertebereich von $-1 \dots +1$.

Neben den sigmoidalen Aktivierungsfunktionen finden in automatisierungstechnischen Anwendungen auch die radialen Basisfunktionen (RBF) Verwendung. Die sog. RBF-Netze konvergieren im Training schneller, haben jedoch die negative Eigenschaft, daß sie durchsacken [9]. Darüberhinaus erfordern RBF-Netze eine höhere Anzahl von Neuronen gegenüber den Multi Layer Perceptrons bzw. Back-Propagation-Netzen.

Mit Hilfe des Back-Propagation Algorithmus erfolgt bei den Multi Layer Perceptrons die Anpassung der Gewichte im Hinblick auf eine gute Übereinstimmung zwischen vorgegebenen Zielgrößen und den Ausgangswerten des Neuronalen Netzes. Die Lern- oder Trainingsphase wird überwacht, und die Auswertung des Fehlervektors bietet die Möglichkeit, den Erfolg des Trainings bewerten zu können.

Neuronale Netze haben in der Automatisierungstechnik sowohl in der Modellierung als auch Steuerung und Regelung Anwendungsbereiche gefunden [1] [6].

Die Möglichkeit, das Wissen über die Strecke anhand von Betriebsdaten den Netzen einprägen zu können, woraus das Netz eigenständig Schlußfolgerungen (z. B. Klassenbildung) ziehen kann, ist als Vorteil dieses neuen Konzeptes anzusehen [1].

Die Erarbeitung von Regeln und/oder eines mathematischen Modells erledigt das Training Neuronaler Netze quasi nebenbei. Ein weiterer Vorteil besteht für die Regelungstechnik z. B. in der Kompensation nichtlinearer Sensorkennlinien oder Identifikation nichtlinearer Strecken [7].

Diese Vorteile stellen sich allerdings nicht ohne menschliche Aktivität ein. Die halbautomatische Trainingsphase, die üblicherweise Voraussetzung für die Nutzung Neuronaler Netze ist, muß als Nachteil angesehen werden:

1. Die Dauer und damit der Aufwand der Trainingsphase ist nicht abschätzbar. Es ist Erfahrungswissen notwendig (Einstellung Lernrate und anderer Parameter).
2. Die Einbeziehung von Expertenwissen in das Training ist schwierig, d. h. dem Netz muß Wissen in Form von Beispieldatensätzen zugeführt werden.
3. Es kann keine Garantie für das Verhalten in nicht trainierten Situationen gegeben werden. (Stabilitätsbeweise Neuronaler Regler gestalten sich schwierig.)
4. Die optimale Struktur (Anzahl Schichten, Neuronen) muß durch Probieren gefunden werden [6].

Die kurz skizzierten Eigenschaften sowie Vor- und Nachteile Neuronaler Netze erfordern einen besonderen Zuschnitt für den in dieser Arbeit vorliegenden neuen Anwendungsfall. Besonders positive Bedeutung haben in diesem Zusammenhang die Fähigkeit als universal approximator und die generalisierenden Eigenschaften. Ein Neuronales Netz wird mit einer endlichen Menge von Mustern trainiert. Aufgrund der generalisierenden Eigenschaften, können Neuronale Netze auch im Fall von nicht trainierten Mustern sinnvolle Ergebnisse ausgeben, z. B. eine Funktion zwischen den trainierten Stützstellen sehr genau approximieren.

Wichtigste Konsequenz aus den nachteiligen Eigenschaften ist die Erarbeitung von mehreren Netzen für unterschiedliche Streckentypen (aperiodisch, periodisch, langsam, schnell). Sinnvoll ist die Erzeugung von Neuronalen Inbetriebnahmenetzen für bestimmte Anwendungsbereiche (Durchfluß-, Temperatur-, Druck-, ... Regelungen). Der Unterschied dieser Netze wird lediglich in den Parametersätzen bestehen.

Das Betragsoptimum

Das Reglerentwurfsverfahren nach dem Betragsoptimum [2] ist für Strecken mit der Übertragungsfunktion

$$F(s) = \frac{1}{\dots + a_3 s^3 + a_2 s^2 + a_1 s + a_0} \quad a_0 \neq 0 \quad (3)$$

anwendbar. Gegenüber dem Verfahren der Polkompensation und Verstärkungseinstellung nach Phasenreserve liefert das Betragsoptimum bei gleicher Überschwingeite ein schnelleres Regelverhalten. Darüber hinaus ist keine Berechnung von Nullstellen des Nennerpolynoms erforderlich. Die Parameter eines PID-Reglers sind aus den Streckenkoeffizienten a_0, a_1, \dots, a_5 direkt berechenbar. Das Verfahren hat sich in der Praxis bewährt und bietet gutes Führungs- und Störverhalten [12].

Der Name des Verfahrens ist insofern irreführend, da kein Optimum im mathematischen Sinne erreicht wird. Es erfolgt keine Berücksichtigung eines Gütekriteriums. Das Optimum besteht darin, daß der Betrag des Frequenzgangs des geschlossenen Kreises für einen möglichst großen Frequenzbereich den Wert eins annimmt.

$$|F_w(j\omega)| \approx 1 \quad (4).$$

Diese Forderung wird gemäß [2] für tiefe Frequenzen näherungsweise erfüllt, wenn ein PI-Regler die folgende Einstellung aufweist:

$$K_R = \frac{a_1}{2} \frac{a_1^2 - a_0 a_2}{a_1 a_2 - a_0 a_3} - \frac{a_0}{2} \quad (5)$$

$$T_n = \frac{a_1}{a_0} - \frac{a_1 a_2 - a_0 a_3}{a_1^2 - a_0 a_2} \quad (6)$$

Die Streckenkoeffizienten können nur durch Identifikation gewonnen werden. Auch im günstigen Fall einer $P - T_n$ -Strecke mit n reellen Polstellen:

$$F(s) = \frac{1}{(1 + T_1 s)(1 + T_2 s) \cdots (1 + T_n s)} \quad (7)$$

wenn $a_0 = 1$ und lediglich a_1 auf einfache meßtechnische Weise über die Steuerfläche bestimmt werden kann ($a_1 = T_\Sigma$), verbleiben mit a_2 und a_3 zwei nicht zu vernachlässigende Koeffizienten. Wird eine aperiodische Strecke, wie im Inbetriebsetzungsgerät SIEPID [12], durch ein Streckenmodell mit n gleichen $P - T_1$ -Gliedern angenähert,

$$F(s) \approx \frac{1}{(1 + T_E s)^n} \quad (8)$$

ist der Reglerentwurf nach dem Betragsoptimum unter Verwendung der Ersatzzeitkonstanten möglich. Bei dem hier vorgestellten Verfahren erfolgt das Training auf die Zielparameter K_R , T_n und T_v aufgrund des Streckenmodells (3). Festzuhalten bleibt der nichtlineare Zusammenhang zwischen Regler- und Streckenparametern. Damit scheidet ein Neuronales Netz mit rein linearen Aktivierungsfunktionen aus.

Neuronaler PID-Reglerentwurf

Bei der Entwicklung einer geeigneten Netzstruktur stand die selbstgestellte Anforderung, mit möglichst wenigen Meßdaten und Neuronen auszukommen, im Mittelpunkt. Darüber hinaus mußte der Tatsache Rechnung getragen werden, daß die Dauer einer Übergangsfunktion keinen Einfluß auf die Anzahl der Eingangsdaten haben sollte.

Daher wird als Eingangsgröße nicht der Wert zu einem bestimmten Zeitpunkt, sondern die Zeit bis zum Erreichen eines bestimmten Wertes verwendet (Zeitprozentkennwert). Der Einsatzbereich ist auf aperiodische Strecken begrenzt.

Dem Netz werden 17 Eingangsgrößen ($T_{5\%}$, $T_{10\%}$, $T_{15\%}$, ..., $T_{95\%}$) zugeführt und zusammen mit den über das Betragsoptimum berechneten Reglerparametern (K_R , T_n , T_v) trainiert. Bild 4 zeigt eine gemessene Übergangsfunktion einer Laborstrecke und einige Prozentlinien.

Die Trainingsstrecken müssen den betrachteten Raum der Streckenzeitkonstanten gleichmäßig abdecken. Nur dann besteht die Aussicht, keine unerwünschten Effekte im Fall von nicht trainierten Eingangsgrößen (und das wird der Einsatzfall sein) entstehen zu lassen. Die Trainingsdatenmenge kann durch ausreichend viele zufällig generierte Streckenzeitkonstanten zusammengestellt werden oder mit Hilfe eines deterministischen Algorithmus, der ein Raster unterschiedlicher Kombinationen von Streckenzeitkonstanten erzeugt.

Mit dem letztgenannten Verfahren wurden die besten Ergebnisse erzielt. Bild 5 symbolisiert einen gleichmäßig aufgespannten Parameterraum für drei Streckenzeitkonstanten. Liegen die Parameter außerhalb des trainierten Parameterraumes, was im Fall aperiodischer Strecken leicht über die Steuerfläche abschätzbar ist, sollte kein Neuronaler Reglerentwurf durchgeführt werden.

Der Parameterraum gemäß Bild 5 enthält redundante Parameterkombinationen. Diese sollten im Hinblick auf das Training durch einen entsprechenden Algorithmus vermieden werden.

Im Rahmen dieser Arbeit wurde von einer Streckenordnung $n = 4$ ausgegangen. Soll ein Inbetriebnahmenetz für Strecken mit einer maximalen Summenzeitkonstanten $T_\Sigma = 20$ s trainiert werden, stellen sich positive Ergebnisse im Fall von jeweils 12 trainierten Streckenzeitkonstanten mit einem Abstand von 0,4 s ein. Die vier Streckenzeitkonstanten nehmen die Werte 0,4 0,8 1,2 ... 4,8 ein ($k = 12$).

Somit entstehen

$$\binom{k+n-1}{n} = \binom{15}{4} = 1365$$

Kombinationen ohne Redundanz bzw. 1365 verschiedene Strecken. Mit Hilfe der Neural Network Toolbox [14] wurde ein MLP-Netz mit einer Zwischenschicht aus 5 Tangens hyperbolicus Neuronen (2) und drei linearen Neuronen für die Ausgangsschicht trainiert. Den simulierten Sprungantworten der 1365 ausgewählten Strecken wurden die Zeitprozentkennwerte entnommen und dem Netz als Eingangsgrößen zur Verfügung gestellt. Die Zielgrößen wurden über das Betragsoptimum vorgegeben.

Ergebnisse Simulation

Es ist verbreitet, die Leistungsfähigkeit neuer regelungstechnischer Verfahren anhand einiger Beispiele zu belegen. Auf der Grundlage einer statistischen Untersuchung mit vielen zufällig ausgewählten Strecken, die den Anwendungsbereich abdecken, können jedoch mehr Informationen über Stärken und Schwächen gewonnen werden.

Die Leistungsfähigkeit dieses Verfahrens wird mit 100 per Zufallszahlengenerator erzeugten Strecken überprüft. Dazu

werden gleichverteilte Zeitkonstanten im Intervall $[0,2 \text{ s } 5 \text{ s}]$ verwendet. Als Maß für den Fehler wird der maximale Abstand f_{MA} zwischen den Übergangsfunktionen des neuronal eingestellten und des auf Basis des bekannten mathematischen Modells eingestellten Reglers definiert. Das Fehlermaß gibt Auskunft über die Auswirkung einer möglichen Parameterabweichung.

Für 100 Zufallsstrecken 4. Ordnung ergibt sich eine Häufigkeitsverteilung des Fehlers gemäß Bild 6 mit einem mittleren Fehler von 1,44% bezogen auf den Endwert. 91 Regelkreise werden mit einem Fehler von kleiner 3% eingestellt. Einstellungen mit einem Fehler $> 5\%$ entstehen im Fall von mehreren sehr kleinen Zeitkonstanten ($T < 0,4 \text{ s}$).

Störungen

Von besonderer Bedeutung für die Beurteilung der Leistungsfähigkeit ist die Einbeziehung von nicht trainierbaren Störungen in Form von Meßrauschen. Eine additive Überlagerung der Übergangsfunktion mit gleichverteilten Zufallszahlen von $\pm 2,5\%$ (bezogen auf den Endwert) verschlechtert erwartungsgemäß die Ergebnisse, siehe Histogramm Bild 7 (mittlerer Fehler 7,27%). Abhilfe könnte ein Glättungsfilter bringen, der als konstantes $P - T_1$ -Glied mit trainiert werden muß.

Niederfrequente sinusförmige Störungen stellen eine besondere Herausforderung für ein Inbetriebnahmewerkzeug dar. Derartige Störungen können nicht mit einem Filter geglättet werden. Erfolgt die Störung wie folgt: $x(t) = \dot{x}(t) + 0,025 \sin(2t)$, dann stellt sich ein mittlerer Fehler von 7,58% ein und eine Häufigkeitsverteilung gemäß Bild 8. Bild 9 zeigt ein besonders schlechtes Beispiel einer Reglereinstellung mit einem Fehler $f_{\text{ma}} = 25\%$. Die zugrundeliegende gestörte Sprungantwort wurde mit dem System

$$F(s) = \frac{1}{(4,8243s + 1)(4,6986s + 1)(1,2748s + 1)(0,3517s + 1)}$$

erzeugt. Die mittlere Kurve zeigt die Einstellung gemäß Betragsoptimum auf Grundlage bekannter Streckenparameter. Der neuronal eingestellte Regelkreis, weist ein zu großes Überschwingen auf.

Zusammenfassung

Im Mittelpunkt dieser Erfindung steht die Verknüpfung der Regeleinrichtung (z. B. PID-Regler) mit dem modernen systemtheoretischen Konzept der Neuronalen Netze. Anwendungszweck ist die Verbesserung der Inbetriebnahme und Wartung von Regelkreisen, wobei im Rahmen dieser Beschreibung eine Einschränkung auf aperiodische Strecken vorgenommen wurde. Diese Einschränkung ist nicht prinzipieller Natur.

Die Streckenübergangsfunktion ist Grundlage eines neuronalen Reglerentwurfes, der auf Basis des Betragsoptimums erfolgt. Die Ergebnisse belegen bereits für ein kleines Netz mit wenigen Eingängen (5 Neuronen, 17 Eingänge) ein gutes Einstellverhalten, auch im Fall von Störungen und nicht trainierten Systemordnungen.

Die Trainingsphase benötigt, wie bei Neuronalen Netzen üblich, einen großen zeitlichen und auch personellen Aufwand. Der Anwendungsfall entspricht jedoch dem Drücken einer Adapt-Taste eines Kompaktreglers und setzt keinerlei Kenntnisse über Neuronale Netze voraus.

Die Lern- und Trainingsphase wurde auf der Basis von simulierten Strecken durchgeführt, da auf diesem Wege ein funktionaler Zusammenhang zwischen Strecke und Reglerparameter sichergestellt ist. Neuronale MLP-Netze lernen Funktionen zu approximieren und sind dann in der Lage zu generalisieren.

Das Reglerentwurfsverfahren ist innerhalb der vorgestellten Struktur frei wählbar. Der Autor hat auch das Verfahren der Polkompensation und Einstellung nach Phasenreserve in dieser Struktur realisieren können.

Literatur

- [1] Anwendungsstand Künstlicher Neuronaler Netze in der Automatisierungstechnik. Gemeinschaftsbeitrag des GMA-Ausschusses "Künstliche Neuronale Netze", Teil 1 – Teil 6, atp 34 (1992) 10 bis atp 35 (1993) 6.
- [2] Keßler, C.: Über die Vorausberechnung optimal abgestimmter Regelkreise Teil III. Regelungstechnik 3 (1955), S. 40–49.
- [3] Klein, M., Walter, H. und Pandit M.: Digitaler PI-Regler: Neue Einstellregeln mit Hilfe der Streckensprungantwort. at 8/92, S. 291–299.
- [4] Kuhn, U.: Eine praxisnahe Einstellregel für PID-Regler: Die T-Summen-Regel. atp 5/95, S. 10–16.
- [5] Latzel, W.: Einstellregeln für vorgegebene Überschwingweiten. at 4/93, S. 103–113.
- [6] Nowinski, G.: Lernfähige konnektionistische Strukturen in der Automatisierungstechnik. at 1/94.
- [7] Otto, P.: Identifikation nichtlinearer Systeme mit Künstlichen Neuronalen Netzen: at 2/95, S. 62–68.
- [8] Pfeiffer, B.-M.: Selbsteinstellende klassische Regler mit Fuzzy-Logik. at 2/94, S. 69–73.
- [9] Preuß, H.-P. und Tresp, V.: Neuro-Fuzzy. atp 5/94, S. 10–24.
- [10] Preuß, H.-P.: Prozeßmodellfreier PID-Regler-Entwurf nach dem Betragsoptimum. at 1/91, S. 15–22.
- [11] Preuß, H.-P.: Robuste Adaption in Prozeßreglern. atp 4/91, S. 178–187.
- [12] Preuß, H.-P., Linzenkirchner, E. und Kirchberg, K.-H.: SIEPID – ein Inbetriebsetzungsgerät zur automatischen Regleroptimierung. atp 9/87, S. 427–436.
- [13] Stenz, R. und Kuhn, U.: Vergleich: Fuzzy-Automatisierung und konventionelle Automatisierung einer Batch-Destillationskolonne. atp 5/93, S. 288–295.
- [14] The Mathworks, Inc.: Neural Network Toolbox. Natick, Mass. 1993.

1. Selbsteinstellungseinrichtung für eine Regeleinrichtung (2) für einen Prozeß (1) mit mindestens einer Stellgröße (y) und einer Regelgröße (x), **dadurch gekennzeichnet**, daß die Selbsteinstellungseinrichtung ein Neuronales Netz (3) aufweist, das in Abhängigkeit von der Regelgröße (x) die Parameter (p) der Regeleinrichtung entwirft.
2. Selbsteinstellungseinrichtung nach Anspruch 1, dadurch gekennzeichnet, daß eine Testfunktion (z. B. Sprung, Rampe oder Dirac) als Stellgröße bei ausgeschalteter Regeleinrichtung oder als Führungsgröße (w) bei eingeschalteter Regeleinrichtung, die Grundlage für die neuronale Selbsteinstellung ist.
3. Selbsteinstellungseinrichtung nach Anspruch 2, dadurch gekennzeichnet, daß die abgetastete Antwort der Regelgröße (x) auf die Testfunktion dem Neuronalem Netz direkt zugeführt wird oder umgerechnet, z. B. in Form von Zeitprozentkennwerten, zugeführt wird.
4. Selbsteinstellungseinrichtung nach Anspruch 3, dadurch gekennzeichnet, daß das Neuronale Netz in einem ersten Schritt auf Grundlage der verwendeten Testfunktion und eines (konventionellen) Reglerentwurfsverfahren trainiert wurde. Dazu ist für eine Klasse von bekannten Prozessen und den dazugehörigen Reglerparametern ein Lern Datensatz zusammenzustellen. Dieser Lern Datensatz enthält die abgetastete Regelgröße als Muster und die Reglerparameter als Zielgröße.

Hierzu 3 Seite(n) Zeichnungen

- Leerseite -

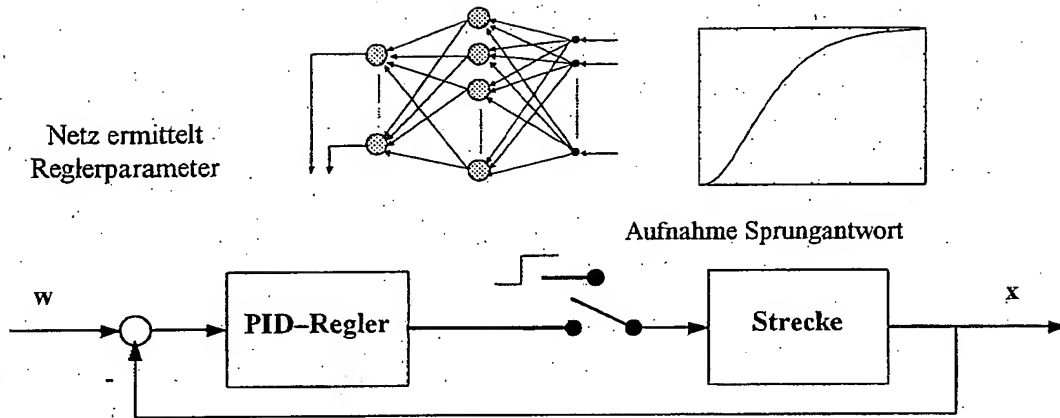


Bild 1: Neuronaler Reglerentwurf

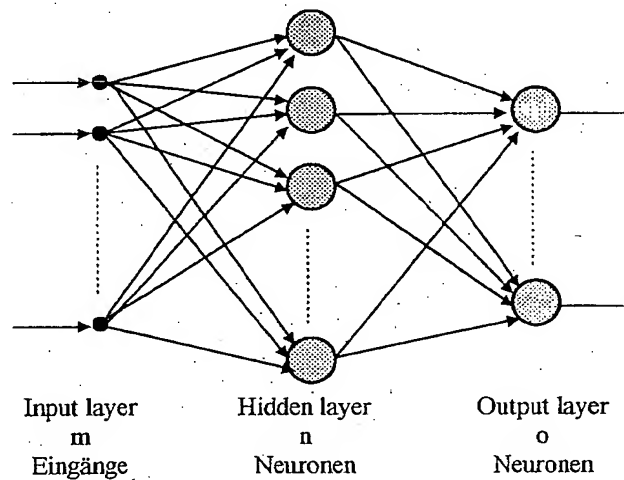


Bild 2: Multi Layer Perceptron

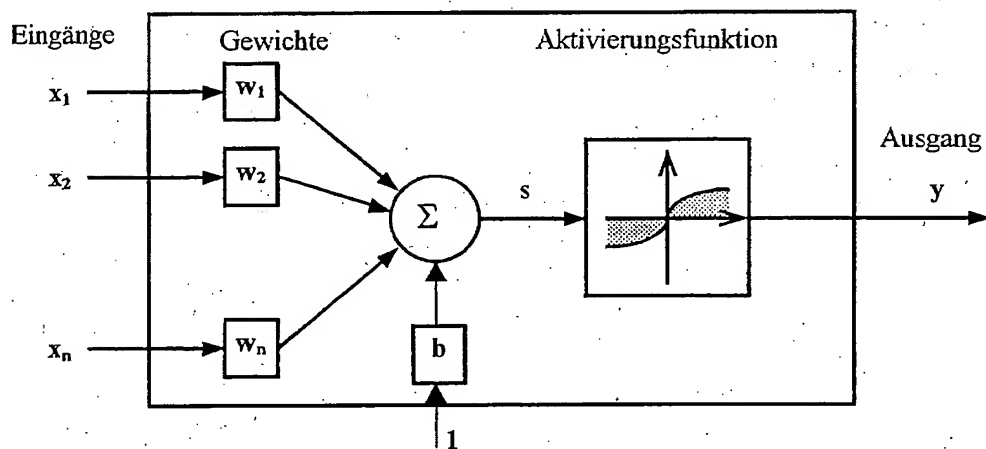


Bild 3: Aufbau eines Neurons

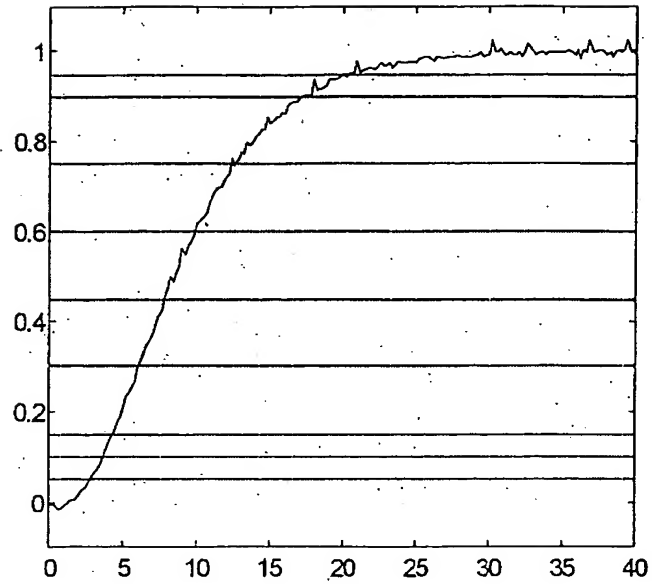


Bild 4: Gemessene Übergangsfunktion mit Prozentlinien

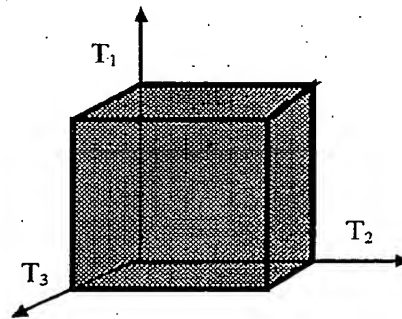


Bild 5: Parameterraum für 3 Zeitkonstanten

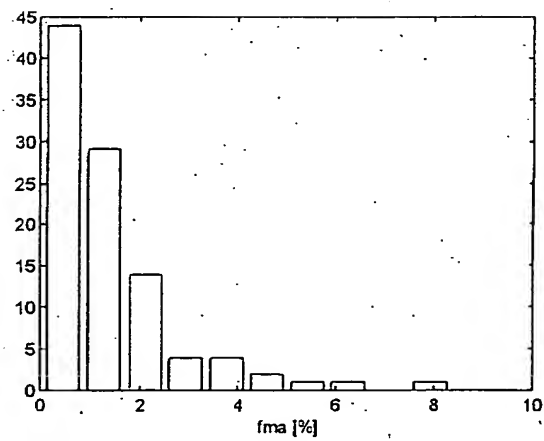


Bild 6. Häufigkeitsverteilung der Fehler

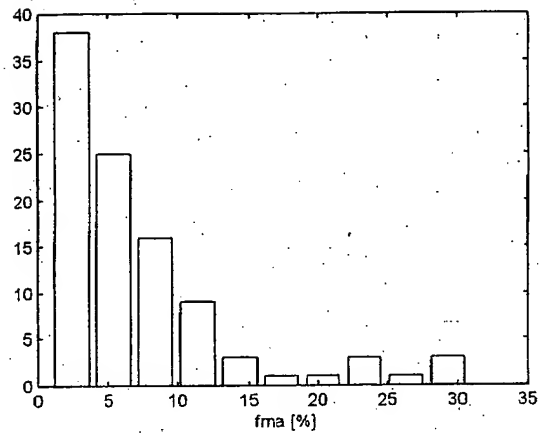


Bild 7. Häufigkeitsverteilung der Fehler im Fall von Störungen / Meßrauschen $\pm 2,5\%$

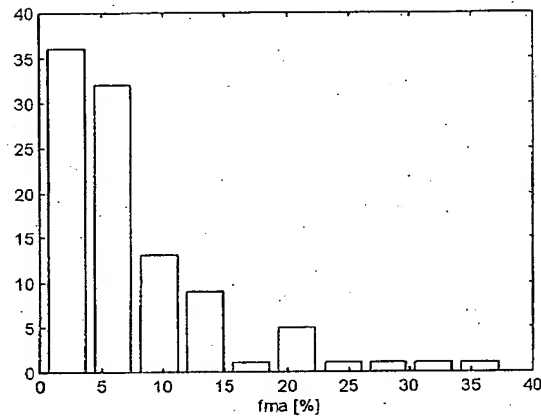


Bild 8. Häufigkeitsverteilung der Fehler im Fall von sinusförmigen Störungen $\pm 2,5\%$

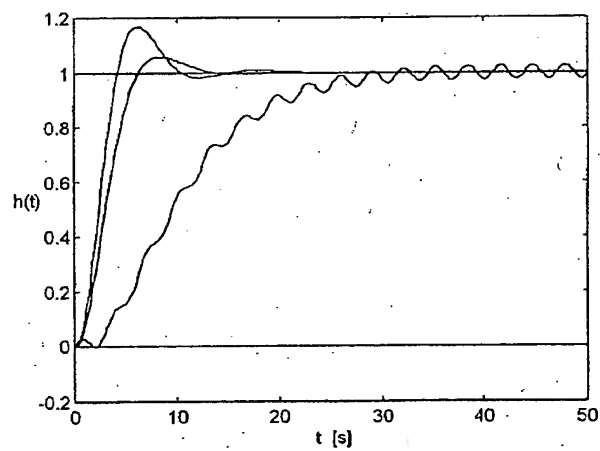


Bild 9. Einstellung mit $f_{ma} = 25\%$

**This Page is Inserted by IFW Indexing and Scanning
Operations and is not part of the Official Record**

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☐ BLACK BORDERS
- ☐ IMAGE CUT OFF AT TOP, BOTTOM OR SIDES
- ☒ FADED TEXT OR DRAWING
- ☒ BLURRED OR ILLEGIBLE TEXT OR DRAWING
- ☐ SKEWED/SLANTED IMAGES
- ☐ COLOR OR BLACK AND WHITE PHOTOGRAPHS
- ☐ GRAY SCALE DOCUMENTS
- ☐ LINES OR MARKS ON ORIGINAL DOCUMENT
- ☐ REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY
- ☐ OTHER: _____

IMAGES ARE BEST AVAILABLE COPY.

As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.

THIS PAGE BLANK (USPTO)